An Approach for Information Discovery Using Ontology In Semantic Web Content

Conference Paper · April 2018

DOI: 10.1145/3209914.3209940

Ayhan Akgun
Bahçeşehir University

Serkan Ayvaz
Bahçeşehir University

An Approach for Information Discovery Using Ontology In Semantic Web Content

Ayhan Akgün
Department of Computer Engineering
Bahcesehir University
Istanbul,Turkey
ayhan.akgun@stu.bahcesehir.edu.tr

Serkan Ayvaz
Department of Software Engineering
Bahcesehir University
Istanbul,Turkey
serkan.ayvaz@eng.bahcesehir.edu.tr

ABSTRACT
Information searching techniques are rapidly developing as the World Wide Web (WWW) evolves. Along with the development of information technologies, the need for acquiring domain knowledge bases, accessing data sources and discovering insights increases. The advancements in knowledge discovery, information management and artificial intelligence require faster data processing, storing more data and developing more intelligent applications. This study provides an information discovery and data integration approach for linked open data in the semantic web. Using semantics embedded in ontologies, data available in knowledge bases can be enhanced to better serve the information needs of users. The entity relationships between resources and resource hierarchies represented as linked open data in semantic web provide semantically rich insights about the data and facilitates knowledge discovery. Graph theory methods can be utilized to enrich the features of data sets in semantic web. In this study, we propose an approach for integrating isolated data sources with semantic web by using ontologies to make them available for information discovery and enhancing the features of semantic data by using graph theory techniques.

CCS Concepts
• Information systems→Network data models→Information systems→Resource Description Framework (RDF) →Information systems→Web Ontology Language (OWL) →Information systems→Ontologies information systems→Information retrieval

Keywords
Graph; information retrieval; ontology; semantic web.

1. INTRODUCTION
From Information Retrieval (IR) perspective, the discovery of related data is a prime challenge when dealing with big data sources. With continuing development of web technologies, data platforms and increased user adoption, IR techniques have gained more importance and played key role in tasks of finding relevant information from large volume of text data in the Web. Information Retrieval involves tasks of storing, indexing, structuring, and searching resources from data collections for obtaining related information. With the advancements of new technologies, it is becoming increasingly important for the IR systems to consider more sophisticated solutions for better information discovery.

Accessing to semantic data is a challenge. Semantic data can be described by means of the information retrieval techniques, approaches and methodologies. The development of standards in machine readable data models as proposed by semantic web can help guide data publishers to publish data in a standard manner. Also, the semantic web data standards, guidelines and the advances in semantic web technologies help support the engineers and content developers to produce content and develop applications conforming semantic data standards.

From information retrieval aspect, data collection, user query formulation and retrieval of ranked data are simply the most important tasks of search engines. On the other hand, the tasks of defining concepts and relations according to standard vocabularies, publishing data in machine-readable formats, providing proper mapping and links for resources are also essential for information discovery and efficient semantic searches. Once properly published and indexed, the semantic web data can be efficiently processed by search systems when searching for relevant information.

In this paper, a new approach is provided that utilizes ontologies to integrate unstructured data repositories with graph structured semantic web content and helps in finding more relevant information in datasets by using graph theory techniques in semantic web data.

The main contributions of this study can be summarized as follows:

- Utilizing information retrieval techniques for finding related information.
- Gathering useful descriptive information from graph data itself for semantic search purposes by utilizing graph theory methods.
- Providing a semantic web approach for converting data in various formats to RDF semantic data model and integrating them into ontologies.

2. BACKGROUND AND RELATED WORK

2.1 Rapid Growth of Data
Data in the Web is increasing rapidly in size in variety of formats. In semantic web data, concepts are linked and organized in

ICISS'18, April 27–29, Jeju, Republic of Korea.

ACM ISBN 978-1-4503-6421-8/18/04…$15.00
https://doi.org/10.1145/3209914.3209940

taxonomies. Having data linked and structured in semantic data model facilitates discovery of relevant information. By providing a framework for publishing and linking data in a standard machine-readable data model, the semantic web promotes data integration and exchange [1]. Describing the resources and the relationships between concepts in semantic data model also make data easily searchable.

While data on the Web is getting bigger every day, the task of searching relevant information is getting more difficult. Searching information only by using keywords may not always provide expected results. On the other hand, there have been many research studies investigating querying information based on ontology for search engines.

In [2], P. N. Gupta et al. proposed an approach that implements a semantic web application focusing on searching information in various datasets. They argue that explicit semantic information on the web pages can be detected by intelligent agents and thus can help resolve issues with complex queries when searching for relevant information in large data collections.

The authors in [3] propose an approach for web search engines using their index metrics and ranking algorithms processed in their stored data, which expose the lack of quality results and inefficient keyword search processing due to misunderstandings in their meaning.

2.2 Information Management in Domain Specific Linked Open Data and Ontologies

There are many data repositories that are used by various domains for information retrieval purposes. Thus, the knowledge bases that are commonly used by different domains need to be again integrated. By mapping all required ontologies and linking data resources, the resources and relationships among resources in data repositories can be used for various purposes and those data collections can be utilized as the linked datasets.

The Linking Open Data (LOD) [4] project is one of the implementations that is driving linked data in semantic web. The purpose of the linked data projects to create connections between distributed semantic web knowledge bases. Another example open-source project Bio2RDF[5] uses semantic web and integration technologies providing linked data environment for medical science. In life sciences domain, there are several noteworthy efforts like Neurocommons [6], Linked LifeData [7], Chem2Bio2RDF [8] that promote linked open data initiatives. In Bio2RDF, the linked data stored in platform can be explored and queried online [9].

For linking and defining information, structured data formats are used, XML structure can help defining information but for more meaningful manner RDF and RDF schema structures can help for defining information in class hierarchies, domain and range definitions. An RDF is Directed-Labeled-Graph (DLG) [10] and it contains nodes which are either resources or literals. The information is provided by using the triple statements (subject-predicate-object) using subject nodes, predicate (URI) and an object node with using URI.

Although resources are addressed by URI and identified as triples, there is still a need to discover the semantics of the resources using ontologies. W3C (World Wide Web Consortium) defines an ontology standard language called OWL (Web Ontology Language) which identifies a number of characteristics and use cases for defining data for semantic web, which provides much more expressiveness than RDF and RDF Schema that offer [11] .While defining and linking data resources are important issues of semantic web, the discovering and accessing the resources are critical challenges in providing linked data solutions [12].

The links between resources in the data graph and their ranks provide important insights about the relationships and the "relatedness" of the resources. One of the first successful implementations of ranking in the web was developed by Google, which treats the web as a graph and utilizes the incoming/outgoing links in the web pages, and the popularity of web pages in search results, among other features in determining the page ranks. Similar techniques are applicable to semantic web searches as well [18].

Graphs represent entities as nodes and the ways in which those entities relate to the world as relationships [13]. In semantic web, both RDF and graph database structures can be used for storing and accessing information. Graph databases provide node and edge definitions for graphs and additional features associated with them [17]. Using graph theory, measuring the similarity between web pages in search engines is possible and similar approaches are used practically in some semantic search engines. The Matchsim proposes a sample solution based on neighbour-based similarity measure algorithm [15]. It defines similarity between web pages by taking the average similarity of the maximum matching between their neighbours using graph algorithms.

The problems of using RDF query languages in graph structure are investigated in [16]. They examined the languges including RQL, RDQL, SeRQL, Triple, N3, Versa and found that these have limited support for graph properties. As RDF graph is considered as directed graph, this directionality causes problems in query languages that do not have a union operator when retrieving neighborhoods. Concerning path finding problem, most languages have support querying for patterns of paths which are limited in length. Aggregated functions like COUNT, MIN, MAX for answering queries about degree of a node, the distance between nodes, and the diameter of a graph are not systematically supported.

From a different angle, a SPARQL [17] query engine G-Store [18] proposed a solution for enriching SPARQL features and address efficient process of SPARQL queries. The solution is based on graph theory and rdf dataset stored as a graph. To achieve better query performance, data indexing is used. However, SPARQL querying has the following limitations for using in the RDF applications:

a. SPARQL queries with wildcards; while querying rdf data, wildcards needs to be supported in query keywords.

b. Dynamic RDF repositories; rdf repositories are updated regularly so that query engines could use these updated datasets.

In this study, we utilize the graph theory techniques to provide insights for finding the patterns in adjacent nodes and edges rather than using major semantic query language SPARQL and Cypher.

3. SAMPLE APPLICATION USING BIOINFORMATICS DATA

3.1 Data Sources and Integration

In this paper, an application infrastructure is proposed for semantic web and statistical analysis of data in graph structure.

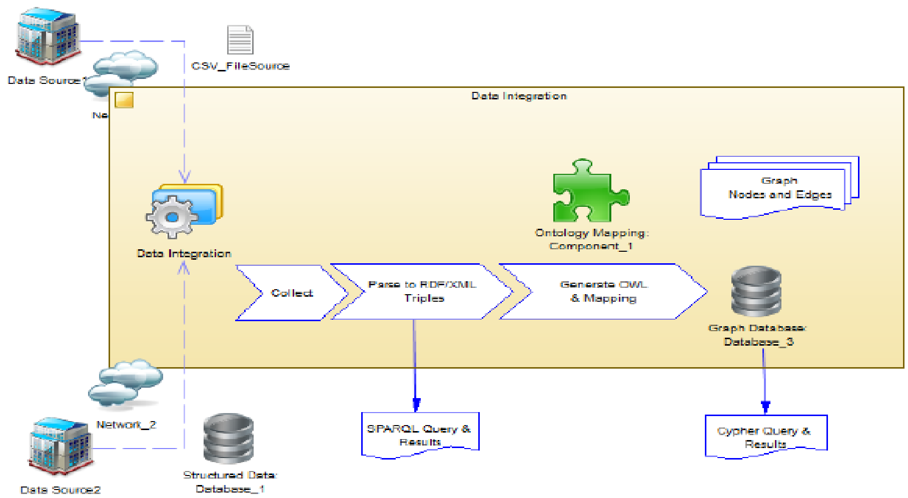

Figure 1. Overview of application modules.

The sample implementation uses ontologies defined in Bio2RDF. The Bio2RDF version of BioPortal contains 107 Open biomedical Ontologies (OBO).

Source dataset is transformed into ontology-based RDF/XML data model. A directed graph is generated for statistical analysis of nodes and edges in graph representation.

Supported sample ontologies in Bio2RDF include ChEBI, Protein Ontology and Gene Ontology. Bio2RDF also supports SPARQL endpoint to make use of data with the coordination of

An objective of this study is to find the connectedness of the drug resources based on their relationships and their neighborhoods in the data graph. Our application collects the data sources from semi structured files stored in csv format. Semi-structured file formats are not easily adaptable to semantic systems for inferring information. To make the dataset as machine-readable, the application converts data source to RDF/XML semantic web serialization. Then, a json [19] file is generated for importing data into the graph database. Json is a popular data format for data exchange and data integration.

Neo4j[20], Jena [21] and Virtuoso [22] are commonly used graph data stores for querying and storing linked data in semantic web [13]. During the evaluations, Neo4j data store was used in this study. It provides graph algorithm functions and supports Cypher query language for using in querying neighbor and path based resources. Due to simple data traversals, finding connected data and relationships between nodes, graph methods have attracted increasing attention from semantic web community. In this research, semi-structured data sources have been processed to order to create a graph structure using a python application.

3.2 Application

The application implemented in python generates rdf graph data from semi structured csv files and integrates the data with the ontology. A property graph has been used for generating graph objects which includes labels, attributes and properties on the graph nodes and edges. Python has packages supporting graph and machine learning algorithms. Networkx [23] is one of the packages containing graph algorithms. Using networkx graph, new information from the linked data can be inferred. Information inferred from nodes and edges can be queried in graph query languages such as cypher.

In converting the data to semantic graph, the application defined each drug definition from source system as a node and an interaction between drugs as an edge in the graph. The node popularity statistics have been measured by the number of neighborhood for any defined drug or node using graph algorithms.

In overview, the application architecture has been designed as NCBO BioPortal so that datasets defined in Bio2RDF is connected to other datasets via named references. It is open source project for provisioning linked data for the life sciences. illustrated in figure 1. The data integration application has the following modules:

a.) Data collection: The drug-drug-interactions data source from Merged PDDI project is collected as a CSV file and imported into the application.

b.) Parse to RDF/XML and OWL serialization: Collected data is converted to RDF/XML. At this stage, data can be stored and queried by RDF databases using SPARQL.

c.) Generation OWL serialization: Collected data is converted to OWL. Ontology classes and individuals are defined according to Bio2RDF ontology. At this stage, defined owl data can be stored in graph database and can be linked to other linked open data sources such Bio2RDF.

d.) Graph structure generation: After generation of RDF, OWL files, a directed graph is generated using json file format as defined figure 3. Json data file is imported to Neo4j database for querying data and visualizing the network map. As another method for searching data in graph, Graph object is generated by using the Python networkx package. Nodes statistics are calculated by using node neighborhood.

3.3 Integrating Data into Ontology

Since there are many data sources available as semi-structured files various domains, it is often necessary that the sources must be consolidated into one domain knowledge for various purposes. Ontologies can be used for integrating related data resources from multiple data sources while searching for related information. Table 1 demonstrates an excerpt of data stored in csv file used in data integration step of the application.

TABLE I. Sample drug interaction data

object_drugbank	object	precipitant_drugbank	precipitant
DB00001	Lepirudin	DB01381	Rilonacept
DB00004	Etanercept	DB00374	Treprostinil

Drug and interaction details can be defined and listed in semantic web information sets in owl formats. Accessing information in semantic datasets requires forming complex queries using a structured query language.

The research dataset collected in this study is from Merged PDDI project [24], which is an open source project containing potential drug-drug interaction data that has been merged into a single data model from 14 different data sources.

The dataset has been transformed into OWL RDF/XML serialization. As shown in figure 2, the drugs have been defined as described definition format in Bio2RDF portal. The class definitions were defined as Drug, Drug-Drug-Interaction and Resource. The ontology vocabulary has the definitions: ddi-interactor-in, Drug, Drug-Drug-Interaction and resource. NamedIndividual defines the each instance of classes in the RDF-OWL data source. The direction of each interaction is defined by ObjectProperties in ontology. Edges are defined by ddi-interactor-in and targets are defined by ddi-interactor-out object property of Drug Resource. Sample drug interaction ontology has been defined in figure 2.

```
<Ontology/>
<ObjectProperty rdf:about="http://bio2rdf.org/drugbank_vocabulary:ddi-interactor-in"/>
<Class rdf:about="http://bio2rdf.org/drugbank_vocabulary:Drug"/>
<Class rdf:about="http://bio2rdf.org/drugbank_vocabulary:Drug-Drug-Interaction"/>
<Class rdf:about="http://bio2rdf.org/drugbank_vocabulary:Resource"/>
<Class rdf:about="http://www.w3.org/2000/01/rdf-schema#Resource"/>
<NamedIndividual rdf:about="http://bio2rdf.org/drugbank:DB00285">
  <drugbank_vocabulary:ddi-interactor-in rdf:resource="http://bio2rdf.org/drugbank_resource:DB00285_DB00503"/>
</NamedIndividual>
<NamedIndividual rdf:about="http://bio2rdf.org/drugbank:DB00503">
  <drugbank_vocabulary:ddi-interactor-in rdf:resource="http://bio2rdf.org/drugbank_resource:DB00285_DB00503"/>
</NamedIndividual>
<NamedIndividual rdf:about="http://bio2rdf.org/drugbank_resource:DB00285_DB00503">
  <rdf:type rdf:resource="http://bio2rdf.org/drugbank_vocabulary:Drug-Drug-Interaction"/>
  <rdf:type rdf:resource="http://bio2rdf.org/drugbank_vocabulary:Resource"/>
  <bio2rdf_vocabulary:identifier rdf:datatype="http://www.w3.org/2001/XMLSchema#string">DB00285_DB00503</bio2rdf_vocabulary:identifier>
  <bio2rdf_vocabulary:namespace rdf:datatype="http://www.w3.org/2001/XMLSchema#string">drugbank_resource</bio2rdf_vocabulary:namespace>
  <bio2rdf_vocabulary:uri rdf:datatype="http://www.w3.org/2001/XMLSchema#string">http://bio2rdf.org/drugbank_resource:DB00285_DB00503</bio2rdf_vocabulary:uri>
</NamedIndividual>
<NamedIndividual rdf:about="http://bio2rdf.org/drugbank_vocabulary:Drug">
  <rdf:type rdf:resource="http://bio2rdf.org/drugbank_vocabulary:Resource"/>
</NamedIndividual>
<NamedIndividual rdf:about="http://bio2rdf.org/drugbank_vocabulary:Drug-Drug-Interaction">
  <rdf:type rdf:resource="http://bio2rdf.org/drugbank_vocabulary:Resource"/>
</NamedIndividual>
<NamedIndividual rdf:about="http://bio2rdf.org/drugbank_vocabulary:Resource">
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Resource"/>
<NamedIndividual rdf:about="http://bio2rdf.org/drugbank_vocabulary:ddi-interactor-in">
  <rdf:type rdf:resource="http://bio2rdf.org/drugbank_vocabulary:Resource"/>
</NamedIndividual>
```

Figure 2. Ontology format in OWL RDF/XML serialization.

Directed Graph object could be generated in Neo4J database with importing data in json file format using the definitions as directed multi graph as shown in figure 3. The node elements represent vertices and each link represents an edge between two nodes. Links are assumed as subclass of class objects. Each class has a representation of one specific drug resource defined according to ontology definition. Node degree and drug-interactions can be observed using the neo4j graph network map. The json file content with nodes and links information is imported to neo4j database using the python-neo4j connection.

```
{
  "directed": false,
  "multigraph": false,
  "graph": {},
  "nodes": [ { "type": "class", "id": "http://ddi.org/drugbank-resource/DB00001","nodeid": 2 },
    { "type": "class", "id": "http://ddi.org/drugbank-resource/DB01381", "nodeid": 3},
    {"type": "class","id": "http://ddi.org/drugbank-resource/DB00374","nodeid": 4},
    {"type": "class", "id": "http://ddi.org/drugbank-resource/DB00004", "nodeid": 5 }],
  "links": [ {"type": "SubClassOf", "source": 2, "target": 3, "drug1id": "http://ddi.org/drugbank-resource/DB00001",
    "drug2id": "http://ddi.org/drugbank-resource/DB01381","object":"Lepirudin","precipitant": "Ginkgo biloba" },
    { "type": "SubClassOf", "source": 2, "target": 4, "drug1id": "http://ddi.org/drugbank-resource/DB00001",
    "drug2id": "http://ddi.org/drugbank-resource/DB00374", "object": "Lepirudin", "precipitant": "Treprostinil" }
  ]
}
```

Figure 3. Directed graph in Json format.

Sample graph node neighbour node degree information is listed in the following Table 2. Using graph theory methods, the measures of node popularity, node degree are calculated by the application. Node degree is determined by the number of edges between the node and its neighbors. Node popularity can be inferred from the node degree since the more number of neighbours that a node has, the more the drug is involved in drug-drug interactions with other drugs.

Table 2. Drug interactions node degree

Node	Degree
http://ddi.org/drugbank-resource/DB00927	6
http://ddi.org/drugbank-resource/DB00467 Degreee: 16	
http://ddi.org/drugbank-resource/DB01072 Degreee: 61	
http://ddi.org/drugbank-resource/DB01066 Degreee: 8	
http://ddi.org/drugbank-resource/DB01167 Degreee: 78	
http://ddi.org/drugbank-resource/DB01026 Degreee: 91	
http://ddi.org/drugbank-resource/DB08864 Degreee: 13	
http://ddi.org/drugbank-resource/DB00471	1
http://ddi.org/drugbank-resource/DB01124 Degreee: 40	
http://ddi.org/drugbank-resource/DB01326	4
http://ddi.org/drugbank-resource/DB00479 Degreee: 34	
http://ddi.org/drugbank-resource/DB00684 Degreee: 61	
http://ddi.org/drugbank-resource/DB00798 Degreee: 34	
http://ddi.org/drugbank-resource/DB00955 Degreee: 32	

As displayed in Table 2, node degrees express the connections between the nodes. The node popularity attribute is calculated for each node demonstrates the degree of connectedness of the node. An excerpt of node popularity of most connected nodes is demonstrated in the table 3. The sorted list of the nodes can be used during querying information about nodes while traversal of the graph database.

Table 3. Node popularity

NodeID	# of Neighbors
http://ddi.org/drugbank-resource/DB00582	256
http://ddi.org/drugbank-resource/DB00976	204
http://ddi.org/drugbank-resource/DB00427	202
http://ddi.org/drugbank-resource/DB00682	184
http://ddi.org/drugbank-resource/DB00726	176

The generated OWL file is explored using the Protégé ontology editor allows visualizing and editing nodes and edge relationships. It has also support for defining class hierarchies and instance definitions for classes. A sample dataset representation of owl ontology in Protégé is visualized in Figure 4.

4. CONCLUSION

With the development of new technologies, user needs for finding and accessing relevant information increases. Knowledge discovery methods in defining information and searching information in semantic manner get more importance. In this study, we proposed a new approach for knowledge discovery in sematic data, which utilizes statistical analysis of information stored in RDF graph data. Moreover, a sample drug-drug interaction dataset that was merged from various data sources in csv files has been converted to RDF semantic data model and integrated into biomedical domain ontology in order to demonstrate the effectiveness and usefulness of the proposed approach. Additionally, graph theory techniques have been utilized to enhance the features of dataset transformed into RDF data model. By doing so the dataset was made available for knowledge discovery and semantic search purposes. Future work will focus on improvement of the query performance of the application by make practical and effective use of index structures.

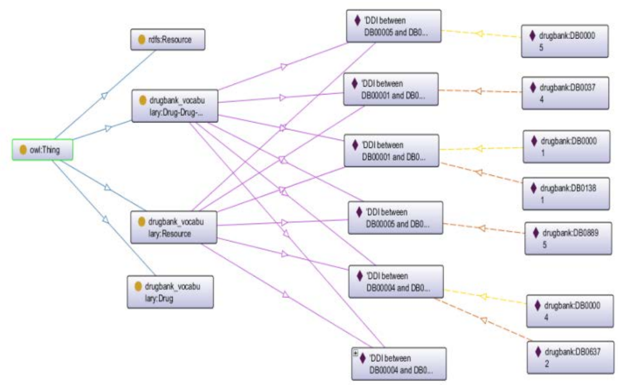

Figure 4. Ontograf schema of drugs interactions graph.

5. REFERENCES

[1] T. Berners-Lee, J. Hendler, and O. Lassila, *The Semantic Web*, vol. 284, no. 5. 2001.

[2] P. N. Gupta, P. Singh, P. P. Singh, P. K. Singh, and D. Sinha, "A Novel Architecture of Ontology based Semantic Search Engine," *Int. J. Sci. Technol.*, vol. 1, no. 12, pp. 650–654, 2012.

[3] S. Kumar, R. a M. K. Rana, and P. Singh, "A Semantic Query Transformation Approach Based on Ontology for Search Engine," vol. 4, no. 5, pp. 688–693, 2012.

[4] S. Yumusak, E. Dogdu, and H. Kodaz, "A short survey of linked data ranking," in *Proceedings of the 2014 ACM Southeast Regional Conference on - ACM SE '14*, 2014, pp. 1–4.

[5] "Linked Data - Design Issues." [Online]. Available: https://www.w3.org/DesignIssues/LinkedData.html. [Accessed: 01-Feb-2018].

[6] A. Ruttenberg, J. A. Rees, M. Samwald, and M. S. Marshall, "Life sciences on the Semantic Web: the Neurocommons and beyond," *Brief. Bioinform.*, vol. 10, no. 2, pp. 193–204, Mar. 2009.

[7] M. Samwald *et al.*, "Linked open drug data for pharmaceutical research and development," *J. Cheminform.*, vol. 3, no. 1, p. 19, May 2011.

[8] B. Chen *et al.*, "Chem2Bio2RDF: a semantic framework for linking and data mining chemogenomic and systems chemical biology data," *BMC Bioinformatics*, vol. 11, no. 1, p. 255, May 2010.

[9] A. Callahan, J. Cruz-Toledo, and M. Dumontier, "Ontology-Based Querying with Bio2RDF's Linked Open Data."

[10] "Semantic Web tutorial - Comparing document formats - slide "Directed Labeled Graph"" [Online].

Available: https://www.w3.org/2003/Talks/0520-www-tf1-a2-formats/slide6-0.html. [Accessed: 12-Jan-2018].

[11] G. Antoniou, F. Van Harmelen, and S. Edition, "A Semantic Web Primer."

[12] A. Reggiori, D. Van Gulik, and Z. Bjelogrlic, "Indexing and retrieving Semantic Web resources : the RDFStore model," *Image (Rochester, N.Y.)*, pp. 1–13, 2010.

[13] I. Robinson, J. Webber, and E. Eifrem, *Graph Databases*. .

[14] C. C. Aggarwal and H. Wang, "MANAGING AND MINING GRAPH DATA."

[15] Z. Lin, M. R. Lyu, and I. King, "MatchSim: A novel similarity measure based on maximum neighborhood matching," *Knowl. Inf. Syst.*, vol. 32, no. 1, pp. 141–166, 2012.

[16] R. Angles and C. Gutierrez, "Querying RDF Data from a Graph Database Perspective."

[17] E. Prud and A. Seaborne, "Sparql query language for rdf," *W3C Recomm. http//www. w3. org/TR/rdf-sparql-query*, no. January 2008, pp. 1–82, 2006.

[18] L. Zou, M. T. Özsu, L. Chen, X. Shen, R. Huang, and D. Zhao, "gStore: A graph-based SPARQL query engine," *VLDB J.*, vol. 23, no. 4, pp. 565–590, 2014.

[19] "JSON." [Online]. Available: https://www.json.org/. [Accessed: 12-Jan-2018].

[20] "The Neo4j Graph Platform – The #1 Platform for Connected Data." [Online]. Available: https://neo4j.com/. [Accessed: 12-Jan-2018].

[21] Apache Software Foundation, "Apache Jena - Home," 2015. [Online]. Available: https://jena.apache.org/. [Accessed: 12-Jan-2018].

[22] OpenLink Software, "OpenLink Virtuoso," 2012. [Online]. Available: http://virtuoso.openlinksw.com/. [Accessed: 12-Jan-2018].

[23] "NetworkX — NetworkX." [Online]. Available: https://networkx.github.io/. [Accessed: 06-Jan-2018].

[24] S. Ayvaz *et al.*, "Toward a complete dataset of drug-drug interaction information from publicly available sources," *J. Biomed. Inform.*, vol. 55, no. April, pp. 206–217, 2015.